



Программа курса

Разработка веб-страниц на языке разметки HTML5 с использованием каскадных таблиц стилей CSS3

Розробка вебсторінок на мові розмітки HTML5 з використанням каскадних таблиц стилів CSS3

Creating Web-pages Using HTML5 and CSS3

Для групп стационара. Версия 1.0.1

Продолжительность курса: 18 пар (9 дней)

Цель курса

Обучить слушателя созданию и верстке статических web-страниц с использованием технологий HTML и CSS. Сложить для слушателя целостное представление о технологической цепочке создания web-сайтов и сформировать понимание актуальных тенденций развития web-технологий. Научить слушателя выбирать наиболее подходящий способ для создания web-страниц. Научить тестировать и проверять код web-страниц.

По окончании курса слушатель будет:

- знать и уметь применять основы HTML – теги, атрибуты и способы структурирования содержимого web-страниц для создания форматированных документов;
- знать и уметь применять основы CSS – значения, списки, цвета, шрифты и другие метрики форматирования;
- владеть навыками проверки и отладки кода web-документов;
- владеть навыками формирования содержимого web-документов для различных экранов – от стандартных браузеров до мобильных устройств;
- владеть навыками быстрого и качественного форматирования сложных web-документов.

В качестве редактора для создания можно использовать любой бесплатный продукт. Например, Notepad++, Microsoft Visual Studio Community.

Оценка по предмету выставляется на основании всех сданных практических и домашних заданий.

Перед началом данного предмета необходимо предоставить студентам доступ к следующим курсам Microsoft Imagine Academy:

- Introduction to Developing Websites Using HTML and CSS (этот курс является подготовкой к экзамену Microsoft Technology Associate (MTA) 98-383);
- HTML5 Application Development Fundamentals (этот курс является подготовкой к экзамену Microsoft Technical Associate exam 98-375).

Также нужно предоставить доступ к официальному курсу Microsoft для подготовки к экзамену Exam 98-375: HTML5 Application Development Fundamentals.

Тематический план

Модуль 1.	Введение в Web-технологии. Структура HTML. Форматирование текста при помощи HTML	2 пары
Модуль 2.	Форматирование при помощи CSS. Списки. CSS отступы и поля	2 пары
Модуль 3.	Графика в web-дизайне. Гиперссылки. Принципы навигации web-сайта	2 пары
Модуль 4.	Анимация в CSS	2 пары
Модуль 5.	Таблицы.	2 пары
Модуль 6.	Позиционирование. Верстка web-страниц блоками.	2 пары
Модуль 7.	Адаптивный дизайн. Flexbox. Grid Layout	2 пары
Модуль 8.	Формы. Фреймы	2 пары
Модуль 9.	Bootstrap. SASS	2 пары

Модуль 1

Введение в Web-технологии. Структура HTML. Форматирование текста при помощи HTML

1. **Введение в предмет.**
2. **Введение в языки разметки. Язык разметки гипертекста HTML.**
 - Internet.
 - Протокол HTTP.
 - Развитие HTML, версии HTML.
 - Вопросы межбраузерной совместимости. Война браузеров.
 - W3C.
3. **Теги – основной элемент структуры HTML. Правила записи тегов и их атрибутов в стандарте HTML. Синтаксические отличия версий HTML.**
4. **Основные ошибки в записях тегов.**
 - Спецификации `<!DOCTYPE HTML>`.
 - Понятие well-formed.
 - Валидация HTML-документов.
 - Прародители HTML: SGML и XML.
5. **Структура HTML документа.**
 - Основные элементы и их назначения.
 - Новые теги задания структуры: `<header>`, `<nav>`, `<section>`, `<article>`, `<aside>`, `<footer>`. Доступность новых тегов в современных браузерах. Отображение новых тегов в устаревших браузерах.
6. **Кодировки страницы и теги `<meta>`.**
 - Применение тега `<meta>` – задание информации о странице (expires, refresh, autor, copyright, keywords, description).
 - Задание кодировки страницы при помощи тега `<meta>`.
 - Символьные подстановки и кодировки.
7. **Классификация тегов: линейные и блочные.**
 - Линейные.
 - Блочные.

8. **Модель форматирования текста: заголовки и абзацы. Элементы <p>, <h1>...<h6>. Выравнивание текста в блочных элементах: атрибут align.**

9. **Классификация тегов: логическое и физическое форматирования.**

- Теги физического форматирования.
- Теги логического форматирования.
- Краткий обзор основных тегов логического форматирования: <abbr>, <acronym>, <cite>, <code>, , <dfn>, <ins>.

10. **Практика: создание простейшей web-страницы.**

Модуль 2

Форматирование при помощи CSS. Списки. CSS отступы и поля

1. **CSS – каскадные таблицы стилей.**

- Введение. Обзор версий. Назначение: HTML служит для задания структуры, CSS – для форматирования.
- Встраивание CSS в HTML при помощи атрибута style.
Правила записи CSS свойств.

2. **Теги без форматирования <div> – блочный, – линейный.**

3. **Аналогия HTML и CSS на примере линейных и блочных тегов.**

4. **Дополнительные свойства CSS для форматирования текста: letter-spacing, line-height, text-indent, text-transform, white-space, word-spacing.**

5. **Использование атрибутов class и id для задания стилей.**

- Создание стилей для тегов, классов, идентификаторов.
- Понятие селекторов. Правило записи селекторов: селектор тегов, селектор классов, селектор идентификаторов, универсальный селектор *.
- Приоритет использования стилей (tag / class / id / style). Повышение приоритета правилом !important.
- Наследуемость стилей. Стандартные значения свойств.
- Отслеживание стилей при помощи средств браузера для разработки.

6. **Использование внешних CSS файлов стилей.**

- Подключение CSS файлов при помощи тега <link> и инструкции @import.
- CSS файлы и кэш браузера.

7. **Практика: форматирование текста при помощи CSS.**

8. Создание списков.

- Неупорядоченные списки: элементы ``, ``.
- Упорядоченные списки: элементы ``, ``.
- Атрибуты `type`, `value`, `start`.

9. Создание вложенных списков.

10. Форматирование списков при помощи CSS.

- Свойства `list-style-type`, `list-style-image`, `list-style-position`.
- Сокращенная запись свойства `list-style`.
- Оформление многоуровневых списков. Вложенные селекторы.

11. Списки определений: элементы `<dl>`, `<dd>`, `<dt>`.

12. Управление отступами и полями.

- Свойство `margin` и его потомки `margin-left`, `margin-top`, `margin-right`, `margin-bottom`.
- Свойство `padding` и его потомки `padding-left`, `padding-top`, `padding-right`, `padding-bottom`.
- Отличие `padding` от `margin` и их назначения.
- Отмена отступов по умолчанию у некоторых тегов: `<body>`, `<h1>...<h6>`, `<p>`.

13. Практика: создание списков.

Модуль 3

Графика в web-дизайне. Гиперссылки. Принципы навигации web-сайта

1. Форматы графических файлов в web.

2. Тег `` и его атрибуты (`src`, `alt`, `width`, `height`, `border`).

- Свойство `border` – аналог атрибута `border`.
- Задание свойств `margin`, `padding`, `border` для изображения.
- Выравнивание изображений на странице при помощи атрибута `align`. Аналог атрибута `align` – свойство `float`.

3. Фон страницы – свойство `background`.

- Задание фона в виде цвета: `background-color`. Обязательное задание фона для элемента `<body>`.
- Задание фона в виде изображения: `background-image`, `background-repeat`, `background-position`, `background-attachment`.
- Изображения и кэш браузера.

4. Общие сведения о гиперссылках.

- Тег <a> и его атрибуты (href, target).
- Эргономика, удобство навигации.

5. Абсолютная и относительная адресация.

- Организация внешних ссылок.
- Организация внутренних ссылок с помощью элемента <a>. Атрибуты id и name.
- Организация «смешанного» перехода (на указанный элемент во внешнем HTML-документе).
- Графические ссылки. Отмена границ у ссылок.

6. Создание меню при помощи структуры списков (,), его форматирование. Свойство display. Преобразование ссылки в блочный элемент.

7. Псевдоклассы.

- Псевдоклассы ссылок: active, hover, link, visited.
- Псевдоклассы для обычных элементов: first-child, first-line, first-letter.

8. CSS свойство cursor.

9. Практика: работа по разработке галереи изображений.

10. Полезные свойства из CSS.

- Работа с фоном: создание градиентов, изменение размеров фона – свойства background и background-size.
- Работа с границами: скругленные края у блоков – свойства border-radius.
- Задание полупрозрачности элементам страниц – свойство opacity.

11. Работа с мультимедиа.

- Вставка видео на странице посредством тега <video>.
- Вставка аудио на странице посредством тега <audio>.
- Создание изображений и анимации посредством тега <canvas>.
- Использование SVG формата.

Модуль 4

Анимация в CSS

1. Функции 2D-трансформации.

- matrix().
- translate(), translateX(), translateY().
- scale(), scaleX(), scaleY().

- rotate().
- skew(), skewX(), skewY().

2. CSS-фильтры.

- blur().
- brightness().
- contrast().
- drop-shadow().
- grayscale().
- hue-rotate().
- opacity().
- saturate().
- sepia().

3. Правило @keyframes.

4. Свойство animation и его составляющие.

Модуль 5

Таблицы

1. Создание простейшей таблицы. Теги <table>, <tr> и <td>.

- Атрибуты border, cellpadding, cellspacing. Их возможные аналоги CSS: border, padding.
- Указание ширины и высоты ячейки: атрибуты width, height. Правила задания ширины и высоты. Аналоги CSS: свойства width, height.
- Выравнивание данных в таблице: атрибуты align и valign. Аналоги CSS: свойства text-align, vertical-align.
- Управление цветом фона и цветом рамок таблицы (отдельной строки, отдельной ячейки).
- Использование изображений в качестве фона таблицы (отдельной строки, отдельной ячейки).

2. Объединение ячеек: атрибуты colspan, rowspan.

3. Теги логического структурирования таблиц: <thead>, <tbody>, <tfoot>. Теги логического группирования столбцов: <colgroup>, <col>.

4. Управление рамками таблицы: атрибуты frame, rules.

5. Практика: создание сложных таблиц.

6. Основы табличной верстки. Пример табличной верстки: ее минусы.

Модуль 6

Позиционирование. Верстка web-страниц блоками

1. Свойство **position**.

- Рассмотрение позиционирования: `relative` и `absolute`.
- Свойства `top`, `left`, `bottom`, `right`.

2. Свойства **visibility**, **overflow**.

3. Практика.

4. Основы верстки блоками. Правила верстки.

- Вложение блоков.
- Задание ширины и высоты блокам при помощи свойства `width` и `height`.
- Обтекание блоков. Отмена обтекания блоков.
- Правила задания отступов и полей.
- Задание минимальной высоты и ширины блока: свойства `min-height`, `min-width`. Задание этих свойств в браузере IE6.
- Выравнивание внутри блоков (`margin`, `text-align`, `line-height`, `position`). Кроссбраузерность выравниваний.

5. Рассмотрение простейших структур страниц.

- Структура фиксированного размера.

6. Резиновая структура. Блоки с отрицательными **margin**.

Модуль 7

Адаптивный дизайн. Flexbox. Grid Layout

1. Что такое адаптивный дизайн?

- Принципы адаптивного дизайна.
- Метатег `viewport`.
- Медиа-запросы.

2. Что такое Flexbox?

- Плюсы и минусы использования Flexbox.
- Базовые термины.
- Основные свойства Flex-контейнера.
- Главная и поперечная ось.

- Многострочная организация блоков во Flex-контейнере.
- Практические примеры использования.

3. Что такое Grid Layout?

- Как создать Grid-контейнер.
- Строка, столбец.
- Позиционирование элементов.
- Grid-линии.
- Функция repeat.
- Практические примеры использования.

Модуль 8

Формы. Фреймы

1. Введение в формы.

2. Управляющие элементы форм.

- Кнопки (отправки, сброса, пр.).
- Флажки.
- Кнопки с зависимой фиксацией (радиокнопки).
- Всплывающие списки.
- Текстовый ввод.
- Выбор файлов.
- Скрытые управляющие элементы.

3. Создание форм при помощи HTML.

- Элемент <form>.
- Элемент <input>.
- Элемент <button>.
- Элементы <select>, <optgroup> и <option>.
- Элемент <textarea>.
- Метки <label>.
- Структура форм: <fieldset> и <legend>.

4. Валидация форм при помощи HTML.

5. Форматирование элементов форм при помощи CSS.

6. Фреймы и их структура (теоретические сведения).

- Тег <iframe>.
- Использование фреймов для подключения внешних ресурсов (YouTube, Google Maps и т.д.).

Модуль 9

Bootstrap. SASS

1. Что такое Bootstrap?

- История создания Bootstrap.
- Цели и задачи Bootstrap.
- Подключение Bootstrap.
- Система сеток.
- Использование различных паттернов Bootstrap.
- Навигация и Bootstrap.
- Использование изображений и видео.
- Скроллинг.
- Анимированные элементы управления.
- Практические примеры использования.

2. Что такое SASS?

- История создания SASS.
- Цели и задачи SASS.
- Инсталляция и подключение SASS.
- Использование SASS.
- Комментарии.
- Переменные.
- CSS расширения.
- Миксины.
- Функции.
- Практические примеры использования.